5. Навчання нейронних мереж

Навчити нейромережу – значить, повідомити їй, про те чого ми від неї очікуємо. Цей процес дуже схожий на навчання дитини алфавіту. Показавши дитині зображення букви *«А»,* ми питаємо його: «яка це буква?» Якщо відповідь невірна, ми повідомляємо дитині ту відповідь, яку ми хотіли б від неї одержати: «це буква А».

Дитина запам'ятовує цей приклад разом з вірною відповіддю, тобто в її пам'яті відбуваються деякі зміни в потрібному напрамку. Ми повторюватимемо процес пред'явлення букв знову і знову доти, коли всі букви твердо запам'ятають. Такий процес називають «навчання з вчителем» (рис. 5.1).

При навчанні мереж ми діємо абсолютно аналогічно. У нас є деяка база даних, що містить приклади (набір рукописних зображень букв).

Демонструючи зображення букви *«А»* на вхід мережі, ми одержуємо від неї деяку відповідь, не обов'язково вірну. Нам відома і вірна (бажана) відповідь – у даному випадку нам хотілося б, щоб на виході з міткою *«А»* рівень сигналу був максимальний. Звичайно як бажаний вихід у задачі класифікації беруть набір (1, 0, 0,...), де 1 стоїть на виході з міткою «А», а 0 – на всій решті виходів.

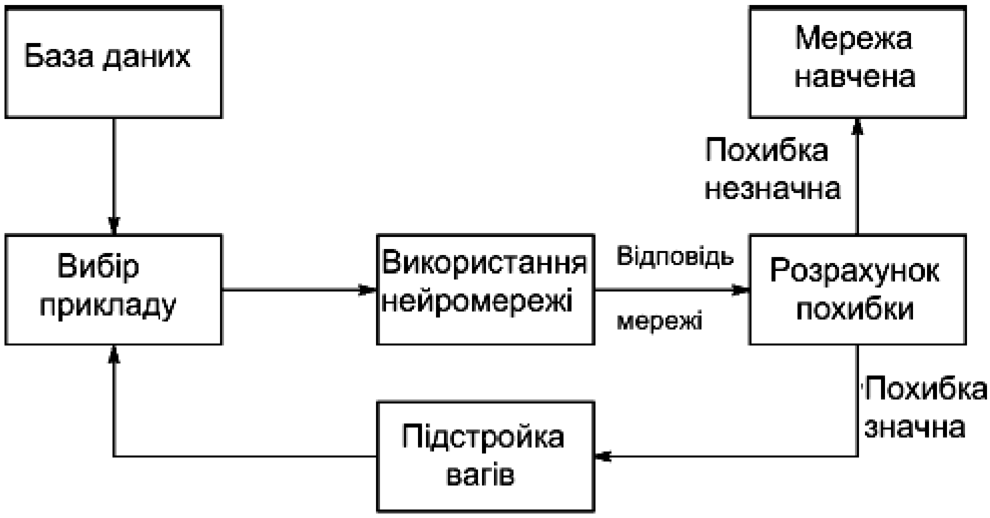

Рис. 5.1. Ілюстрація процесу навчання НМ

Вирахувавши різницю між бажаною відповіддю і реальною відповіддю мережі, ми одержуємо 33 числа – вектор помилки.

Алгоритм навчання – це набір формул, який дозволяє по вектору помилки обчислити необхідні поправки для ваг мережі.

Одну і ту ж букву (а також різні зображення однієї і тієї ж букви) ми можемо пред'являти мережі багато разів. У цьому значенні навчання швидше нагадує повторення вправ у спорті – тренування.

Виявляється, що після багатократного пред'явлення прикладів вага мережі стабілізується, причому мережа дає правильні відповіді на всі (або майже всі) приклади з бази даних. У такому разі говорять, що «мережа вивчила всі приклади», «мережа навчена», або «мережа натренована». У програмних реалізаціях можна бачити, що у процесі навчання функція помилки (наприклад, сума квадратів помилок по всім виходах) поступово зменшується. Коли функція помилки досягає нуля або прийнятного малого рівня, вправи зупиняють, а одержану мережу вважають натренованою і готовою до використання на нових даних.

Важливо відзначити, що вся інформація, яку мережа має про задачу, міститься в наборі прикладів. Тому якість навчання мережі напряму залежить від кількості прикладів у навчальній вибірці, а також від того, наскільки повно ці приклади описують дану задачу. Так, наприклад, безглуздо використовувати мережу для прогнозу фінансової кризи, якщо в навчальній вибірці кризи не представлено. Вважається, що для повноцінного тренування потрібні хоча б декілька десятків (а краще сотень) прикладів.

Математично процес навчання можна описати таким чином.

У процесі функціонування нейронна мережа формує вихідний сигнал Y відповідно до вхідного сигналу X, реалізовуючи деяку функцію $Y = G(X)$. Якщо архітектура мережі задана, то вид функції G визначається значеннями синаптичних вагів і зсувів мережі.

Нехай рішенням деякої задачі є функція $Y = F(X)$, задана парами вхідних-вихідних даних (X^1, Y^1), (X^2, Y^2) ,..., (X^N, Y^N), для яких $Y^k = F(X^k)$ $(k = 1, 2,..., N)$.

Навчання полягає у пошуці (синтезі) функції G, близької до F у значенні деякої функції помилки E (рис. 5.1).

Якщо вибрані множини навчальних прикладів – пара (X^k, Y^k) (де $k=1,2,..., N$) і спосіб обчислення функції помилки E, то навчання нейронної мережі перетворюється на задачу багатовимірної оптимізації, що має дуже велику розмірність, при цьому, оскільки функція E може мати довільний вигляд, навчання у загальному випадку – *багатоекстремальна неопукла задача оптимізації.*

Для вирішення цієї задачі можуть бути використані наступні (ітераційні) алгоритми:

– алгоритми локальної оптимізації з обчисленням часткових похідних першого порядку;

– алгоритми локальної оптимізації з обчисленням часткових похідних першого і другого порядку;

– стохастичні алгоритми оптимізації;

– алгоритми глобальної оптимізації.

До першої групи відносяться: градієнтний алгоритм (метод найшвидшого спуску); методи з одновимірною і двовимірною оптимізацією цільової функції у напрямі антиградієнта; метод зв’язаних градієнтів; методи, що враховують напрям антиградієнта на декількох кроках алгоритму.

До другої групи відносяться: метод Ньютона, квазіньютонівські методи, метод Гауса-Ньютона, метод Льовенберга-Марквардта і інші.

Стохастичними методами є: пошук у випадковому направленні, імітація відпалу, метод Монте-Карло (чисельний метод статистичних випробувань).

Задачі глобальної оптимізації розв'язуються за допомогою перебору значень змінних, від яких залежить цільова функція (функція помилки E).